



(12) **EUROPEAN PATENT APPLICATION**
published in accordance with Art. 153(4) EPC

(43) Date of publication:
11.02.2015 Bulletin 2015/07

(51) Int Cl.:
G06F 13/10 ^(2006.01) **G06F 3/06** ^(2006.01)
G06F 13/14 ^(2006.01) **G06F 13/36** ^(2006.01)

(21) Application number: **13771839.1**

(86) International application number:
PCT/JP2013/002316

(22) Date of filing: **03.04.2013**

(87) International publication number:
WO 2013/150792 (10.10.2013 Gazette 2013/41)

(84) Designated Contracting States:
AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO RS SE SI SK SM TR
Designated Extension States:
BA ME

- **HIDAKA, Youichi**
Tokyo 108-8001 (JP)
- **YASUDA, Masato**
Tokyo 108-8001 (JP)
- **YOSHIKAWA, Takashi**
Tokyo 108-8001 (JP)
- **HIGUCHI, Junichi**
Tokyo 108-8001 (JP)

(30) Priority: **06.04.2012 JP 2012087888**

(71) Applicant: **NEC Corporation**
Tokyo 108-8001 (JP)

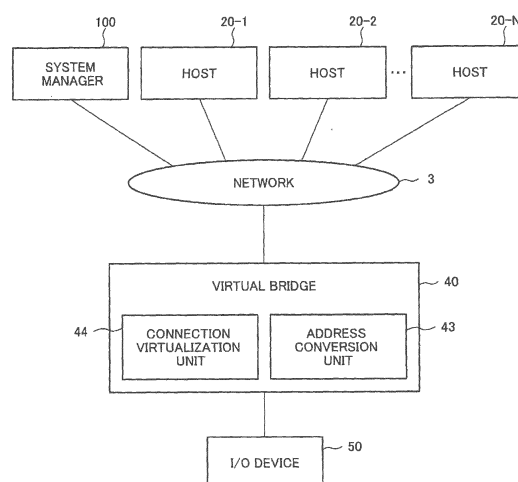
(74) Representative: **Betten & Resch**
Theatinerstrasse 8
80333 München (DE)

(72) Inventors:
• **SUZUKI, Jun**
Tokyo 108-8001 (JP)

(54) **I/O DEVICE SHARING SYSTEM AND I/O DEVICE SHARING METHOD**

(57) An I/O device sharing system characterized by comprising: an I/O device (50) shared by a plurality of hosts (20-1 to 20-N); a system manager (10) which sets the I/O device (50); a virtual bridge (40) which virtualizes the I/O device (50); and a network (3) which connects the I/O device (50), the system manager (10), the plurality of hosts (20-1 to 20-N) and the virtual bridge (40) to each other, wherein the virtual bridge (40) includes a connection virtualization unit (41) by which it is detected that an address setting of a plurality of virtual functions provided in the I/O device (50) that is set by the system manager (10) is performed, the virtual function is enabled, or both of them are performed and each host is permitted to access each virtual function.

Fig. 6



Description

[Technical Field]

[0001] The present invention relates to an Input/Output (I/O) device sharing system and an I/O device sharing method and in particular, relates to an I/O device sharing system which can simplify management of an I/O device and an I/O device sharing method.

[Background Art]

[0002] One example of the I/O device sharing system is described in PTL 1. The I/O device sharing system shown in Fig. 7 includes a plurality of hosts (hereinafter, these hosts are described as a host 2-1 (a host 1), a host 2-2 (a host 2),..., and a host 2-N (a host N)), an I/O device 5 corresponding to SR-IOV, an I/O virtual bridge 6, and a network 3. The I/O device 5 corresponding to SR-IOV conforms to the Single-Root I/O Virtualization (SR-IOV) standard shared by the hosts 2-1 to 2-N. The I/O virtual bridge 6 virtualizes the I/O device 5 corresponding to SR-IOV. The network 3 connects the hosts 2-1 to 2-N and the I/O virtual bridge 6.

[0003] The host 2-1 includes a CPU 201-1, a memory 203-1, a host PCIe bridge 204-1, and a route complex 202-1. The CPU 201-1 operates according to a program. The memory 203-1 stores the program and data. The host PCIe bridge 204-1 connects a PCI Express (PCIe) bus that is an I/O bus of the host 2-1 and the network 3. The route complex 202-1 connects the CPU 201-1, the memory 203-1, and the host PCIe bridge 204-1 to each other. The host PCIe bridge 204-1 encapsulates a PCIe packet in a network packet and decapsulates the PCIe packet encapsulated in the network packet. All the hosts 2-1 to 2-N have the same configuration.

[0004] The I/O device 5 corresponding to SR-IOV holds a Physical Function (PF) 501 that is an interface for controlling the device and a plurality of Virtual Functions (VFs) that are interfaces to provide the function of the I/O device. Hereinafter, a plurality of the VFs are described as the VF 502-1 (VF1) to the VF 502-N (VFN). The PF 501 holds a PF configuration register (PF_CFG_REG) 5011 for setting the PF 501. The VF 502-1 is used by the host 2-1. Similarly, the VFs 502-2 to 502-N are used by the hosts 2-2 to 2-N, respectively. Accordingly, the I/O device 5 corresponding to SR-IOV is shared by the hosts 2-1 to 2-N.

[0005] The I/O virtual bridge 6 includes a network connection unit 601 which transmits/receives the PCIe packet to/from the network 3, a packet transfer unit 602 which transfers the PCIe packet, an address conversion unit 603 which performs conversion of a destination address of the PCIe packet, and a connection virtualization unit 604 which performs a setting of the I/O device 5 corresponding to SR-IOV.

[0006] The network connection unit 601 decapsulates the PCIe packet that is encapsulated in the packet of the

network 3. The network connection unit 601 encapsulates the PCIe packet in the packet of the network 3 and transmits it to the network 3.

[0007] The connection virtualization unit 604 accesses the PF_CFG_REG 5011 and performs the setting of the I/O device 5 corresponding to SR-IOV before the I/O device 5 corresponding to SR-IOV is used by the hosts 2-1 to 2-N. At this time, the address of the I/O device 5 corresponding to SR-IOV is set.

[0008] The address conversion unit 603 converts the address of the PCIe packet with respect to the access from the hosts 2-1 to 2-N to the I/O device 5 corresponding to SR-IOV. Specifically, the address conversion unit 603 performs the address conversion in which the addresses of the VFs 502-1 to 502-N that are set by the hosts 2-1 to 2-N are converted into the addresses of the VFs 502-1 to 502-N that are set by the connection virtualization unit 604, respectively.

[0009] The I/O device sharing system having the above-mentioned configuration operates as follows.

[0010] In the following explanation, a case in which the host 2-1 accesses the VF 502-1 will be explained as an example. The following explanation can be applied to a case in which the hosts 2-2 to 2-N access the VFs 502-2 to 502-N, respectively. When the CPU 201-1 which operates according to software in the host 1 issues an instruction to access the VF 502-1, the route complex 202-1 creates the PCIe packet addressed to a memory address to which the VF 502-1 is mapped and transmits it. The host PCIe bridge 204-1 encapsulates the PCIe packet in the network packet and transmits it to the network 3. The network connection unit 601 receives the network packet, decapsulates the PCIe packet, and transfers it to the packet transfer unit 602.

[0011] The packet transfer unit 602 transfers the received PCIe packet to the address conversion unit 603. The address conversion unit 603 converts the destination address designated by the host 2-1 into the address of the VF 502-1 that is set to the PF 501 by the connection virtualization unit 604 and transmits the PCIe packet to the VF 502-1. As a result, the VF 502-1 receives the PCIe packet issued by the host 2-1. Further, when the host 2-1 receives the PCIe packet transmitted by the VF 502-1, the above-mentioned processes are performed in reverse order.

[Citation List]

[Patent Literature]

[0012] PTL 1: International publication No. 2009/025381

[Summary of Invention]

[Technical Problem]

[0013] However, the system described in PTL 1 has a

problem in which management of the I/O device that is performed by the I/O virtual bridge is complicated. The reason is because each I/O device has a specific setting process and in order to respond to the various I/O devices, the specific setting process of each I/O device needs to be realized by the I/O virtual bridge.

[0014] An object of the present invention is to provide an I/O device sharing system which can simplify management of an I/O device that is performed by an I/O virtual bridge and an I/O device sharing method.

[Solution to Problem]

[0015] According to the present invention, there is provided an I/O device sharing system characterized by comprising: an I/O device shared by a plurality of hosts; a system manager which sets the I/O device; a virtual bridge which virtualizes the I/O device; and a network which connects the I/O device, the system manager, the plurality of hosts, and the virtual bridge to each other, wherein the virtual bridge includes a connection virtualization unit by which it is detected that an address setting of a plurality of virtual functions provided in the I/O device that is set by the system manager is performed, the virtual function is enabled, or both of them are performed and each host is permitted to access each virtual function.

[0016] According to the present invention, there is provided an I/O device sharing method characterized by comprising: connecting an I/O device and a network via a virtual bridge which virtualizes the I/O device, connecting a system manager which sets the I/O device and a plurality of hosts which share the I/O device with the network, and detecting that an address setting of a plurality of virtual functions held by the I/O device that is set by the system manager is performed, the virtual function is enabled, or both of them are performed and permitting each host to access each virtual function.

[Advantageous Effect of Invention]

[0017] The present invention has an effect in which management of the I/O device that is performed by the I/O virtual bridge can be simplified. The reason is because in the management of the I/O device, only the management applied to all the I/O devices is performed by the I/O virtual bridge and the other management that is used for a process individually performed by each I/O device is performed by a system manager. Namely, the process performed by each I/O device excluding the common process is performed by the system manager connected through a network by using a software provided by a manufacturer of the I/O device.

[Brief description of Drawings]

[0018]

[Fig. 1] Fig. 1 is a block diagram showing a configuration of one exemplary embodiment of an I/O device sharing system according to the present invention.

[Fig. 2] Fig. 2 is a state machine diagram showing a state transition of a connection virtualization unit.

[Fig. 3] Fig. 3 is a figure for explaining an address setting performed by a connection virtualization unit.

[Fig. 4] Fig. 4 is a flowchart showing an operation of a case in which a system manager issues a configuration access to a PF.

[Fig. 5] Fig. 5 is a flowchart showing an operation of a case in which a system manager performs a memory access to an area in which a PF is memory-mapped.

[Fig. 6] Fig. 6 is a block diagram showing a main portion of an I/O device sharing system according to the present invention.

[Fig. 7] Fig. 7 is a block diagram showing a configuration of an I/O device sharing system described in PLT 1.

[Description of Embodiments]

[0019] An exemplary embodiment of the present invention will be described in detail below with reference to the drawing.

[0020] Fig. 1 is a block diagram showing a configuration of one exemplary embodiment of an I/O device sharing system according to the present invention. The I/O device sharing system shown in Fig. 1 includes a system manager 10, hosts 2-1 to 2-N, an I/O device 5 corresponding to SR-IOV, an I/O virtual bridge 4, and a network 3. The system manager 10 controls the I/O device 5 corresponding to SR-IOV. The hosts 2-1 to 2-N share the I/O device 5 corresponding to SR-IOV. The I/O virtual bridge 4 connects the I/O device corresponding to SR-IOV and the network 3. The network 3 connects the system manager 10, the hosts 2-1 to 2-N, and the I/O virtual bridge 4 to each other.

[0021] The system manager 10 includes a CPU 101, a memory 103, a host PCIe bridge 104, a route complex 102, and a PF setting software 105. The CPU 101 operates according to software. The memory 103 stores a program and data. The host PCIe bridge 104 connects a PCIe bus and the network 3. The route complex 102 connects the CPU 101, the memory 103, and the host PCIe bridge 104 to each other. The PF setting software 105 controls a PF 501 of the I/O device 5 corresponding to SR-IOV.

[0022] Further, the PF 501 corresponds to one example of a control function for controlling a virtual function.

[0023] The PF setting software 105 includes a device driver which operates in kernel space, an OS, and a PF setting user software which operates in user space. The PF setting user software is generally provided by a manufacturer which provides the I/O device 5 corresponding to SR-IOV as a series of software programs which control the device together with the device. The PF setting user software may be provided by the operating system which

operates in the system manager 10.

[0024] The I/O device 5 corresponding to SR-IOV holds the PF 501 that is an interface which controls the device and VFs 502-1 to 502-N that are the interfaces which provide a function of the I/O device. The PF 501 holds a PF_CFG_REG5011 for setting the PF 501. The VF 502-1 holds a VF configuration register (VF CFG REG) 5021-1 (VF1 CFG REG) for setting the VF 502-1. All the VFs 502-1 to 502-N have the same configuration. The VF 502-1 is used by the host 2-1. Similarly, the VFs 502-2 to 502-N are used by the hosts 2-2 to 2-N, respectively. Accordingly, the I/O device 5 corresponding to SR-IOV is shared by the hosts 2-1 to 2-N.

[0025] The I/O virtual bridge 4 includes a network connection unit 401, a packet transfer unit 402, an address conversion unit 403, a connection virtualization unit 404, and a bridge memory 405. The network connection unit 401 transmits/receives the PCIe packet to/from the network 3. The packet transfer unit 402 transfers the PCIe packet. The address conversion unit 403 performs conversion of the destination address of the PCIe packet. The connection virtualization unit 404 performs a setting of the I/O device 5 corresponding to SR-IOV. The bridge memory 405 stores the setting of the I/O device 5 corresponding to SR-IOV.

[0026] The network connection unit 401 decapsulates the PCIe packet encapsulated in the packet of the network 3. The network connection unit 401 encapsulates the PCIe packet in the packet of the network 3 and transmits it to the network 3.

[0027] The bridge memory 405 includes a virtual PF_CFG_REG 4051, a virtual VF1 CFG REG 4052-1 to a virtual VFN CFG REG 4052-N, and an I/O setting storage unit 4053. The virtual PF_CFG_REG 4051 simulates the PF_CFG_REG 5011. The virtual VF1 CFG REG 4052-1 to the virtual VFN CFG REG 4052-N simulate the VF1 CFG REG 5021-1 to the VFN CFG REG 5021-N, respectively. The I/O setting storage unit 4053 stores the setting of the I/O device 5 corresponding to SR-IOV that is performed by the connection virtualization unit 404.

[0028] The connection virtualization unit 404 operates according to a state machine shown in Fig. 2.

[0029] An activation initial state S1 is a first state in which the I/O virtual bridge 4 is activated. At this time, the I/O virtual bridge 4 reads a parameter which specifies the operation of the connection virtualization unit 404 from a setting storage medium. The state of the I/O virtual bridge 4 changes to an I/O device setting state S2.

[0030] In the I/O device setting state S2, the connection virtualization unit 404 accesses the PF_CFG_REG 5011 of the I/O device 5 corresponding to SR-IOV that is set by the system manager 10, sets the addresses of the PF 501 and the VFs 502-1 to 502-N, and enables the VFs 502-1 to 502-N.

[0031] The address setting performed by the connection virtualization unit 404 will be described with reference to Fig. 3. The connection virtualization unit 404 creates an I/O device memory space M1. In the space M1, the

connection virtualization unit 404 maps the PF 501 in a PF map area M101. Further, the connection virtualization unit 404 maps the VFs 502-1 to 502-N in a VF1 map area M102-1 to a VFN map area M102-N, respectively. Namely, the connection virtualization unit 404 sets the address to the virtual function and the control function. The register used for setting this map is held in the PF_CFG_REG 5011. The connection virtualization unit 404 stores the setting value of the PF 501 in the I/O setting storage unit 4053. The state of the connection virtualization unit 404 changes to a system manager connection state S3.

[0032] In the system manager connection state S3, the access to the PF 501 that is performed by the system manager 10 is permitted. Specifically, the connection virtualization unit 404 makes the network connection unit 401 establish the connection between the host PCIe bridge 104 and the network connection unit 401 so that the system manager 10 can access the PF 501.

[0033] When the connection between the system manager 10 and the I/O device 5 corresponding to SR-IOV is established, software operated by the system manager 10 issues the configuration access to the I/O device 5 corresponding to SR-IOV and performs the configuration. The configuration access includes a configuration write access and a configuration read access by the PCIe packet. The configuration access of the system manager 10 is transferred to the connection virtualization unit 404 by the packet transfer unit 402.

[0034] Next, the connection virtualization unit 404 writes the setting of the system manager 10 to the virtual PF_CFG_REG 4051. By this access, the system manager 10 maps the PF 501 in a PF virtual map area M201 of a system manager physical memory space M2. Namely, the system manager 10 sets the address of the control function for controlling the virtual function. In the configuration access to the virtual PF_CFG_REG 4051 that is performed by the system manager 10, when it is detected by the connection virtualization unit 404 that the VFs 502-1 to 502-N are enabled, the state of the connection virtualization unit 404 changes to a host connection possible state S4. When it is detected by the connection virtualization unit 404 that the address setting of the VFs 502-1 to 502-N is performed or the address setting of them is performed and the VFs 502-1 to 502-N are enabled, the state of the connection virtualization unit 404 may change to the host connection possible state S4.

[0035] The host connection possible state S4 is a state in which the hosts 2-1 to 2-N can use the VF 502-1 to 502-N, respectively. The function with respect to the host 2-1 of the connection virtualization unit 404 will be described below. Similarly, the function of the hosts 2-2 to 2-N can be described.

[0036] The connection virtualization unit 404 establishes the connection between the network connection unit 401 and the host PCIe bridge 204-1 and controls the network connection unit 401 so that the host 2-1 can communicate with the VF 502-1. After the connection is es-

tablished, first, the host 2-1 issues the configuration access to the VF1 CFG REG 5021-1 and performs the configuration of the VF 502-1 (VF1). The configuration access issued by the host 2-1 is transferred to the connection virtualization unit 404 by the packet transfer unit 402. The connection virtualization unit 404 writes the setting of the host 2-1 to the virtual VF1 CFG REG 4052-1. By this setting, as shown in Fig. 3, the host 2-1 maps the VF 502-1 in a VF1 virtual map area M301-1 of a host 1 physical memory space M3-1. Namely, the host 2-1 performs the address setting to the virtual function. As a result, the host 2-1 can perform the memory access to the VF 502-1. Further, the host 2-1 can use the VF 502-1.

[0037] The host connection possible state S4 is a steady state for providing a service in which the I/O device 5 corresponding to SR-IOV is shared by the hosts 2-1 to 2-N. Unless an event such as a system error, a release of the connection between the host PCIe bridge 104 and the network connection unit 401, or the like occurs, the state of the connection virtualization unit 404 is maintained to the host connection possible state S4. When the connection between the host PCIe bridge 104 and the network connection unit 401 is released, the state of the connection virtualization unit 404 changes to an I/O device reset state S5.

[0038] In the I/O device reset state S5, the connection virtualization unit 404 controls the network connection unit 401 so that the connections between the network connection unit 401 and the hosts 2-1 to 2-N are released (the connections are cut off). The connection virtualization unit 404 resets the I/O device 5 corresponding to SR-IOV. After the reset, the state of the connection virtualization unit 404 changes to the I/O device setting status S2, the process is returned to the process in which the setting of the I/O device 5 corresponding to SR-IOV is performed, and the above-mentioned processes are performed again.

[0039] The address conversion unit 403 performs conversion of the address described in the header of the PCIe packet transmitted and received between the system manager 10 and the PF 501. The address conversion unit 403 performs conversion of the address described in the header of the PCIe packet transmitted and received between the hosts 2-1 to 2-N and the VFs 502-1 to 502-N used by the hosts 2-1 to 2-N, respectively. For example, as shown in Fig. 3, the PF 501 is mapped in the PF map area M101 of the I/O device memory space M1. Further, the PF 501 is mapped in the PF virtual map area M201 of the system manager physical memory space M2. In that case, the address conversion unit 403 performs the address conversion corresponding to a difference between map values of these two address spaces. Similarly, the VF 502-1 is mapped in the VF1 map area M102-1 of the I/O device memory space M1. Further, the VF 502-1 is mapped in the VF1 virtual map area M301-1 of the host 1 physical memory space M3-1. In this case, the address conversion unit 403 performs the address conversion of these two address spaces. Similarly, the ad-

dress conversion unit 403 performs the address conversion with respect to the VFs 502-2 to 502-N.

[0040] When the address conversion unit 403 performs the address conversion of the PCIe packet passing through the address conversion unit 403, the address conversion unit 403 refers to information of the I/O device memory space M1, the system manager physical memory space M2, and the host 1 physical memory space M3-1 to the host N physical memory space M3-N that are stored in the bridge memory 405. These information are obtained from the information stored in the virtual PF_CFG_REG 4051, the virtual VF1 CFG REG 4052-1 to the virtual VFN CFG REG 4052-N, and the I/O setting storage unit 4053.

[0041] The I/O device sharing system according to this exemplary embodiment operates as follows based on the above-mentioned configuration.

[0042] First, operation of a case in which the system manager 10 issues the configuration access to the PF 501 will be described with reference to Fig. 1 and Fig. 4.

[0043] The configuration access is the configuration write request or the configuration read request. When the PF setting software 105 operated by the system manager 10 issues the configuration access to the PF 501 (Step A1), the PCIe packet is created by the route complex 102. The PCIe packet is transmitted to the host PCIe bridge 104.

[0044] The host PCIe bridge 104 encapsulates the PCIe packet in the packet of the network 3 and transmits it to the network 3 (Step A2). The network connection unit 401 receives the packet in which the PCIe packet is encapsulated. The network connection unit 401 decapsulates the network packet and transmits it to the packet transfer unit 402 (Step A3). The packet transfer unit 402 confirms that the PCIe packet is the configuration access by the system manager 10 and transfers it to the connection virtualization unit 404 (Step A4). When it is determined that the request is the write request based on the received configuration access, the connection virtualization unit 404 writes the setting value into the virtual PF_CFG_REG 4051. When it is determined that the request is the read request, the connection virtualization unit 404 reads the setting value designated from the virtual PF_CFG_REG 4051 and transmits a reply to the system manager 10 (Step A5).

[0045] Next, operation of a case in which the system manager 10 performs the memory access to the area in which the PF 501 is memory-mapped will be described with reference to Fig. 1 and Fig. 5.

[0046] The memory access is the memory write request or the memory read request. When the PF setting software 105 which operates by the system manager 10 issues the memory access to the PF 501 (Step B1), the PCIe packet addressed to the PF virtual map area M201 is created by the route complex 102. The PCIe packet is transmitted to the host PCIe bridge 104. The host PCIe bridge 104 encapsulates the PCIe packet in the packet of the network 3 and transmits it to the network 3 (Step

B2). The network connection unit 401 receives the network packet in which the PCIe packet is encapsulated, decapsulates the PCIe packet, and transmits it to the packet transfer unit 402 (Step B3).

[0047] The packet transfer unit 402 confirms that the PCIe packet is the memory access by the system manager 10 and transfers the PCIe packet to the address conversion unit 403 (Step B4).

[0048] The address conversion unit 403 performs conversion of the destination address of the PCIe packet from the PF virtual map area M201 to the PF map area M101 by using the value stored in the virtual PF_CFG_REG 4051 which holds the setting value of the PF virtual map area M201 and the value stored in the I/O setting storage unit 4053 which holds the setting value of the PF map area M101 in the bridge memory 405. The address conversion unit 403 transmits the PCIe packet to the I/O device 5 corresponding to SR-IOV (Step B5). As a result, the PF 501 receives the memory access issued by the system manager 10. When the memory access is the memory write request, the PF 501 writes the designated data and when it is the memory read request, the PF 501 reads the designated data and transmits a reply to the system manager 10 (Step B6).

[0049] Fig. 6 is a block diagram showing a main portion of the I/O device sharing system according to the present invention. As shown in Fig. 6, the I/O device sharing system includes an I/O device 50 (for example, the I/O device 5 corresponding to SR-IOV shown in Fig. 1), a system manager 100 (for example, the system manager 10 shown in Fig. 1), a virtual bridge 40 (for example, the I/O virtual bridge 4 shown in Fig. 1), and the network 3.

[0050] The I/O device 50 is shared by a plurality of hosts 20-1 to 20-N (for example, the hosts 2-1 to 2-N shown in Fig. 1). The system manager 100 sets the I/O device 50. The virtual bridge 40 virtualizes the I/O device 50. The network 3 connects the I/O device 50, the system manager 100, a plurality of the hosts 20-1 to 20-N, and the virtual bridge 40 to each other. The virtual bridge 40 includes a connection virtualization unit 44 (for example, the connection virtualization unit 404 shown in Fig. 1). When it is detected by the connection virtualization unit 44 that the address setting of a plurality of the virtual functions provided in the I/O device 50 that is set by the system manager 100 is performed, the virtual function is enabled, or both of them are performed, the connection virtualization unit 44 permits each of the hosts 20-1 to 20-N to access each virtual function.

[0051] The virtual bridge 40 further includes an address conversion unit 43. With respect to the address described in the packet of the I/O bus that is transmitted and received between each of the hosts 20-1 to 20-N and each of the virtual functions, the address conversion unit 43 performs the address conversion by referring to the address that is set to each virtual function by each of the hosts 20-1 to 20-N and the address that is actually set to each virtual function.

[0052] As described above, this exemplary embodi-

ment has a configuration in which the I/O virtual bridge controls the PF_CFG_REG that is common to all the I/O devices and the system manager connected to the I/O device through the network controls the memory map area of the PF that depends on the individual I/O device by using the PF setting software provided by the manufacturer of the I/O device. Therefore, the management of the I/O device that is performed by the I/O virtual bridge can be simplified. Further, a configuration of a portion related to the management of the I/O device that is performed by the I/O virtual bridge can be simplified.

[0053] Further, in this exemplary embodiment, the bus based on the standard of the bus which conforms to the PCIe is used as the I/O bus. However, a bus based on a new I/O bus standard that will be developed by expanding the function of the PCIe in the future or a bus based on another I/O bus standard which operates in a similar manner can be used for the present invention.

[0054] In this exemplary embodiment, the number of the hosts 2-1 to 2-N is equal to the number of the VFs 502-1 to 502-N of the I/O device 5 corresponding to SR-IOV. However, the number of the hosts may not be necessarily equal to the number of the VFs. For example, an exemplary embodiment in which the number of VFs is greater than the number of the hosts and each host uses one of the VFs may be used.

[0055] The I/O device 5 corresponding to SR-IOV according to this exemplary embodiment is a device which conforms to the PCIe SR-IOV standard such as a network interface card, a storage device, a storage network card, a USB controller, and an I/O device for built-in apparatus.

[0056] The invention of the present application has been described above with reference to the exemplary embodiment and the example. However, the invention of the present application is not limited to the above mentioned exemplary embodiment and example. Various changes in the configuration or details of the invention of the present application that can be understood by those skilled in the art can be made without departing from the scope of the invention.

[0057] This application claims priority based on Japanese Patent Application 2012-087888 filed on April 6th, 2012, the disclosure of which is hereby incorporated by reference in its entirety.

[Industrial applicability]

[0058] The present invention can be applied to a use in which a single I/O device is shared by a plurality of computers in a computer system. Further, the present invention can be applied to a use in which a single I/O device is shared by a plurality of apparatuses in a built-in system.

[Reference signs List]

[0059]

2-1 to 2-N host
 3 network
 4 I/O virtual bridge
 5 I/O device corresponding to SR-IOV
 10 system manager
 20-1 to 20-N host
 40 virtual bridge
 43 address conversion unit
 44 connection virtualization unit
 50 I/O device
 100 system manager
 101 CPU
 102 route complex
 103 memory
 104 host PCIe bridge
 105 PF setting software
 201-1 to 201-N CPU
 202-1 to 201-N route complex
 203-1 to 203-N memory
 204-1 to 204-N host PCIe bridge
 401 network connection unit
 402 packet transfer unit
 403 address conversion unit
 404 connection virtualization unit
 405 bridge memory
 501 PF
 502-1 to 501-N VF
 601 network connection unit
 602 packet transfer unit
 603 address conversion unit
 604 connection virtualization unit

Claims

1. An I/O device sharing system **characterized by** comprising:

an I/O device shared by a plurality of hosts;
 a system manager which sets the I/O device;
 a virtual bridge which virtualizes the I/O device;
 and
 a network which connects the I/O device, the
 system manager, the plurality of hosts, and the
 virtual bridge to each other,
 wherein the virtual bridge includes a connection
 virtualization unit by which it is detected that an
 address setting of a plurality of virtual functions
 provided in the I/O device that is set by the sys-
 tem manager is performed, the virtual function
 is enabled, or both of them are performed and
 each host is permitted to access each virtual
 function.

2. The I/O device sharing system described in claim 1, wherein the virtual bridge further comprises an address conversion unit which performs an address conversion by referring to an address that is set to

each virtual function by each host and an address that is actually set to each virtual function with respect to the address described in a packet of an I/O bus that is transmitted and received between each host and each virtual function.

3. The I/O device sharing system described in claim 2 wherein the connection virtualization unit performs the address setting of the virtual function and the address setting of a control function for controlling the virtual function provided in the I/O device, the virtual bridge includes a bridge memory which stores the address that is set to the virtual function and the control function by the connection virtualization unit, the address that is set to the control function by the system manager, and the address that is set to each virtual function by each host, and the address conversion unit performs the address conversion by referring to the address that is set to the control function by the system manager and the address that is set to the control function by the connection virtualization unit with respect to the address described in the packet of an I/O bus that is transmitted and received between the system manager and the control function.

4. The I/O device sharing system described in any one of claim 1 to claim 3 wherein, when the connection virtualization unit detects disconnection of the connection between the system manager and the virtual bridge, the connection virtualization unit cuts off the connections between a plurality of the hosts and the virtual bridge to stop a use of a plurality of the virtual functions by a plurality of the hosts and resets the I/O device.

5. The I/O device sharing system described in any one of claim 1 to claim 4 wherein the I/O device conforms to a standard of PCI Express Single-Root I/O Virtualization.

6. An I/O device sharing method **characterized by** comprising:

connecting an I/O device and a network via a
 virtual bridge which virtualizes the I/O device,
 connecting a system manager which sets the
 I/O device and a plurality of hosts which share
 the I/O device with the network, and
 detecting that an address setting of a plurality
 of virtual functions held by the I/O device that is
 set by the system manager is performed, the
 virtual function is enabled, or both of them are
 performed and permitting each host to access
 each virtual function.

7. The I/O device sharing method described in claim 6 wherein an address conversion is performed by re-

ferring to an address that is set to each virtual function by each host and an address that is actually set to each virtual function with respect to the address described in a packet of an I/O bus that is transmitted and received between each host and each virtual function. 5

8. The I/O device sharing method described in claim 7 wherein the virtual bridge performs the address setting of the virtual function and the address setting of a control function for controlling the virtual function provided in the I/O device, the virtual bridge holds the address that is set to the virtual function and the control function, the address that is set to the control function by the system manager, and the address that is set to each virtual function by each host in a bridge memory, and the address conversion is performed by referring to the address that is set to the control function by the system manager and the address that is set to the control function by the virtual bridge with respect to the address described in the packet of the I/O bus that is transmitted and received between the system manager and the control function. 10 15 20 25
9. The I/O device sharing method described in any one of claim 6 to claim 8 wherein, when the virtual bridge detects disconnection of the connection between the system manager and the virtual bridge, the virtual bridge cuts off the connections between a plurality of the hosts and the virtual bridge to stop a use of a plurality of the virtual functions by a plurality of the hosts and resets the I/O device. 30
10. The I/O device sharing method described in any one of claim 6 to claim 9 wherein the I/O device conforms to a standard of PCI Express Single-Root I/O Virtualization. 35

40

45

50

55

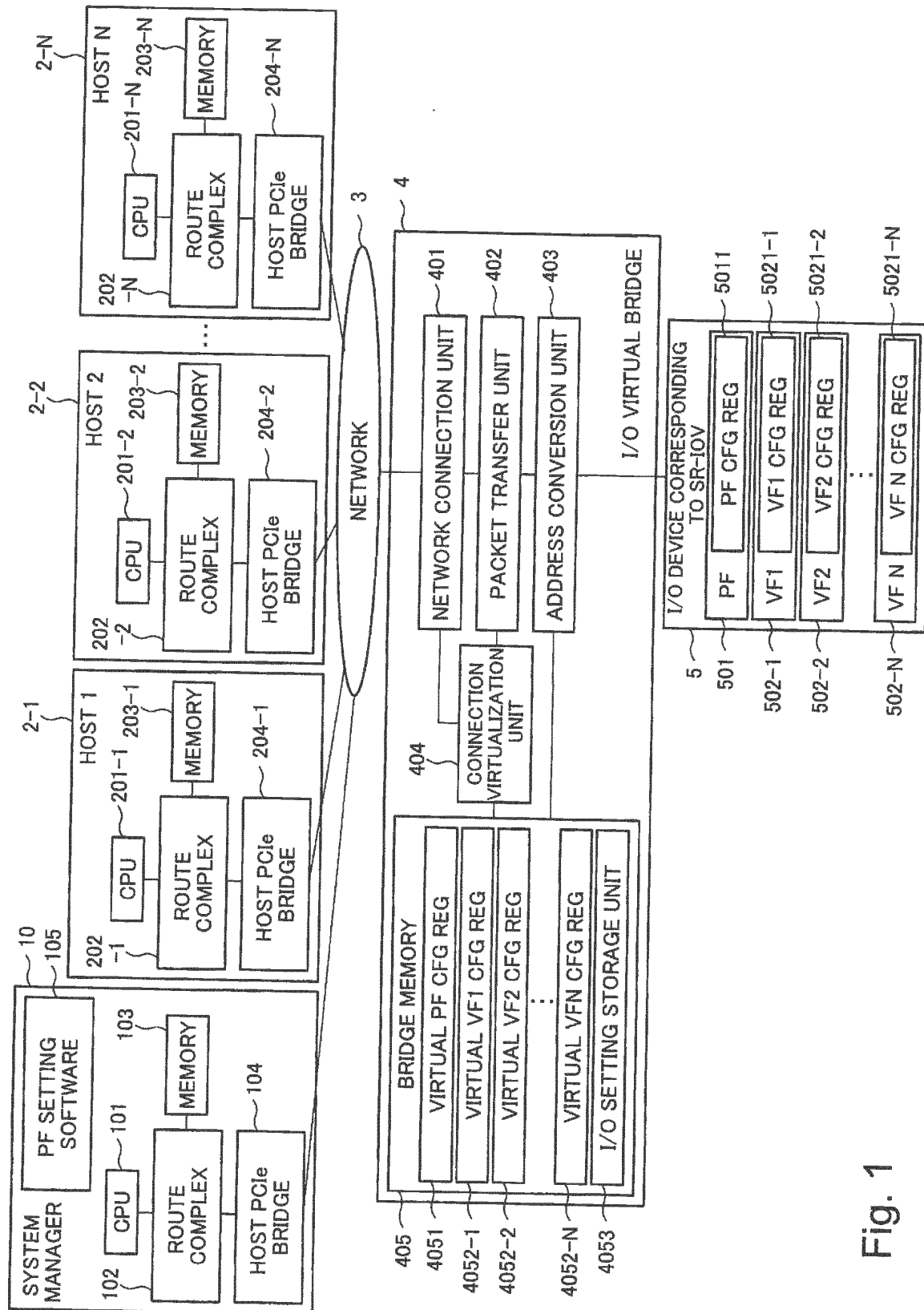
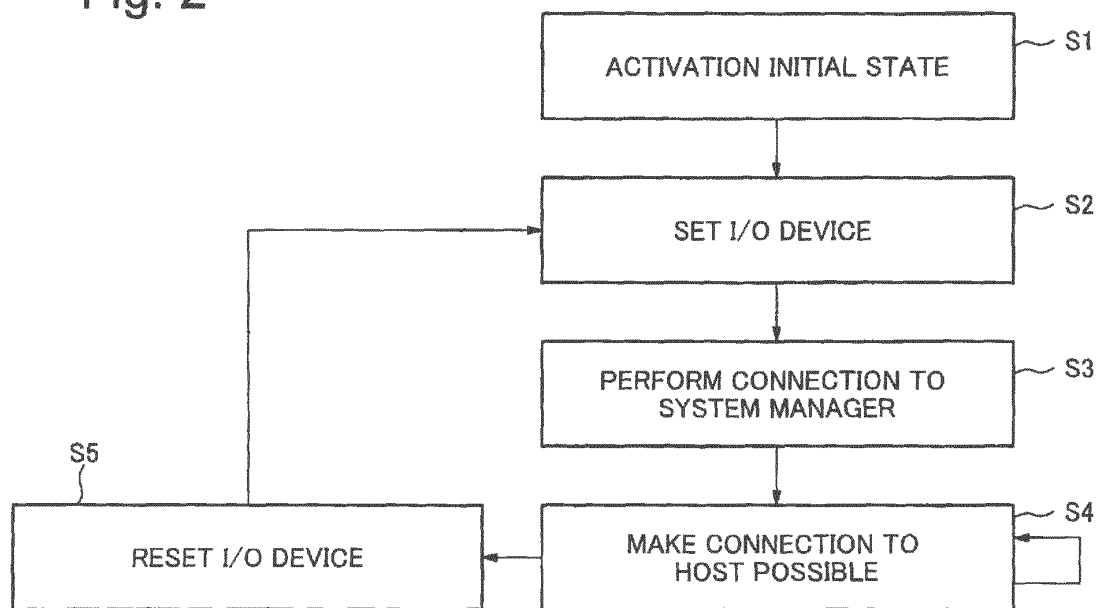


Fig. 1

Fig. 2



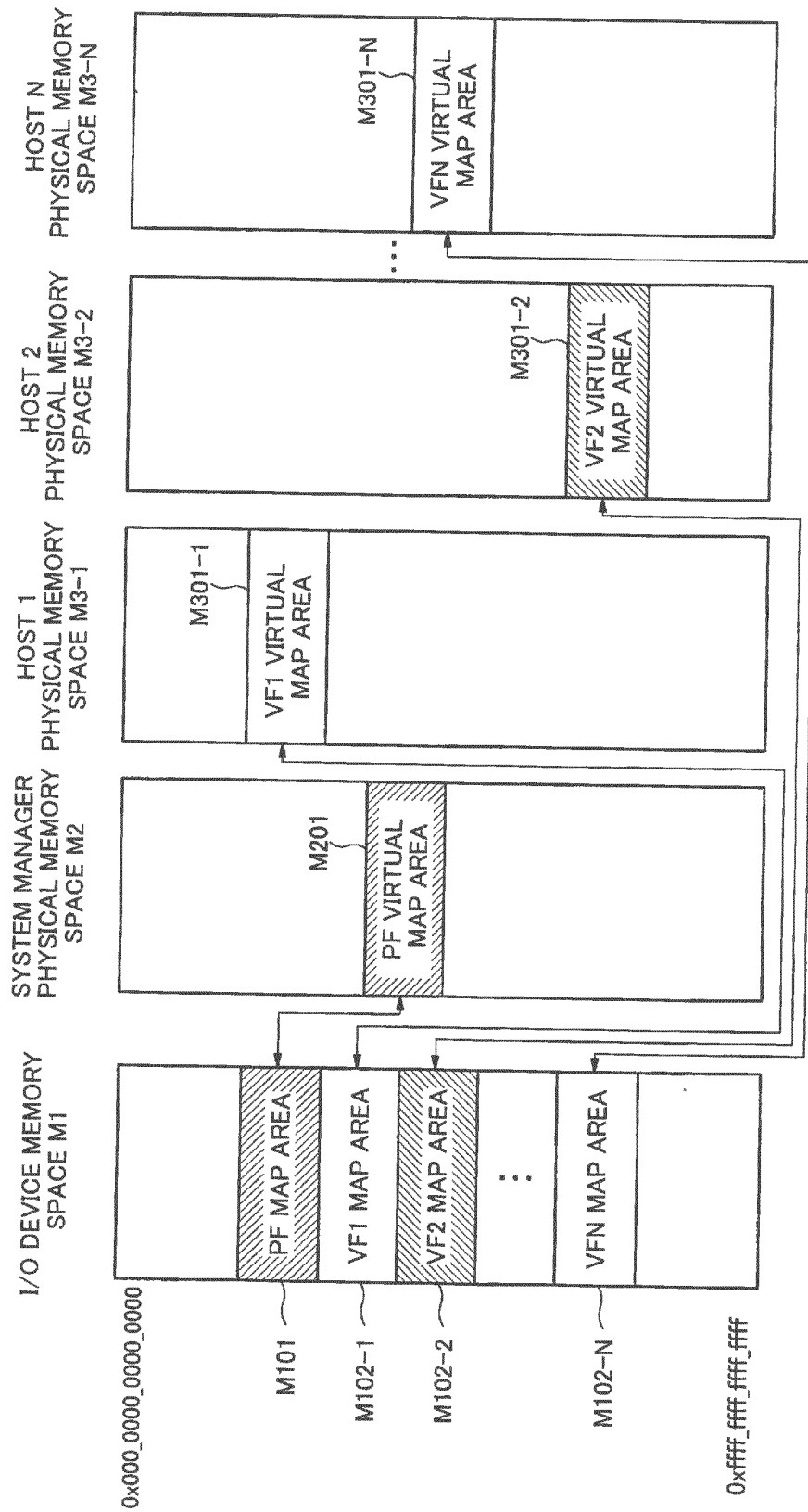


Fig. 3

Fig. 4

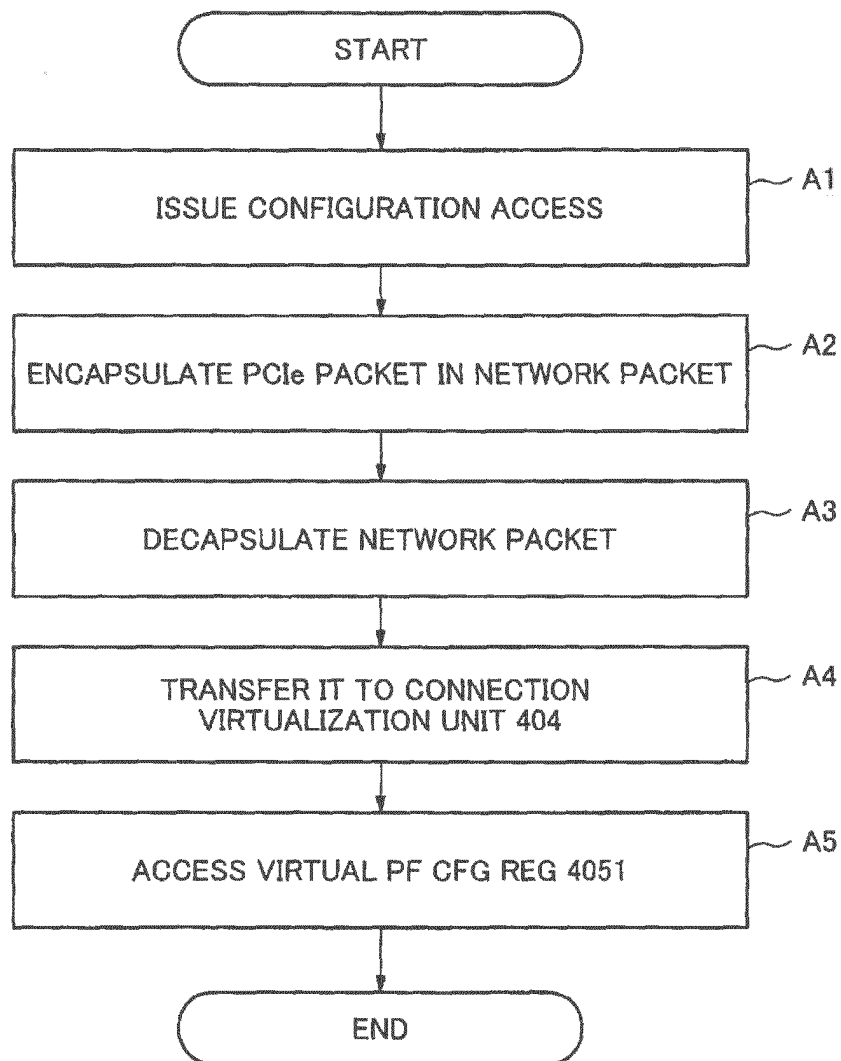


Fig. 5

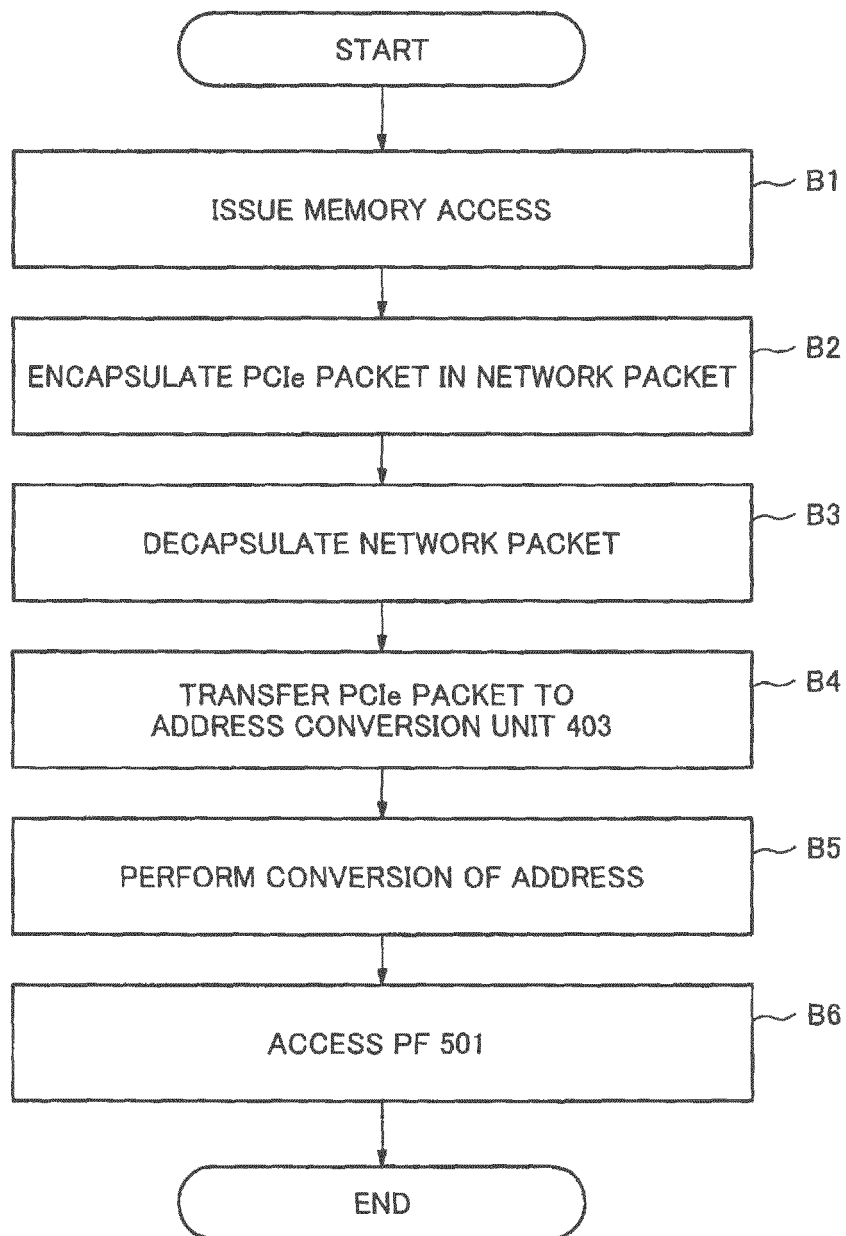
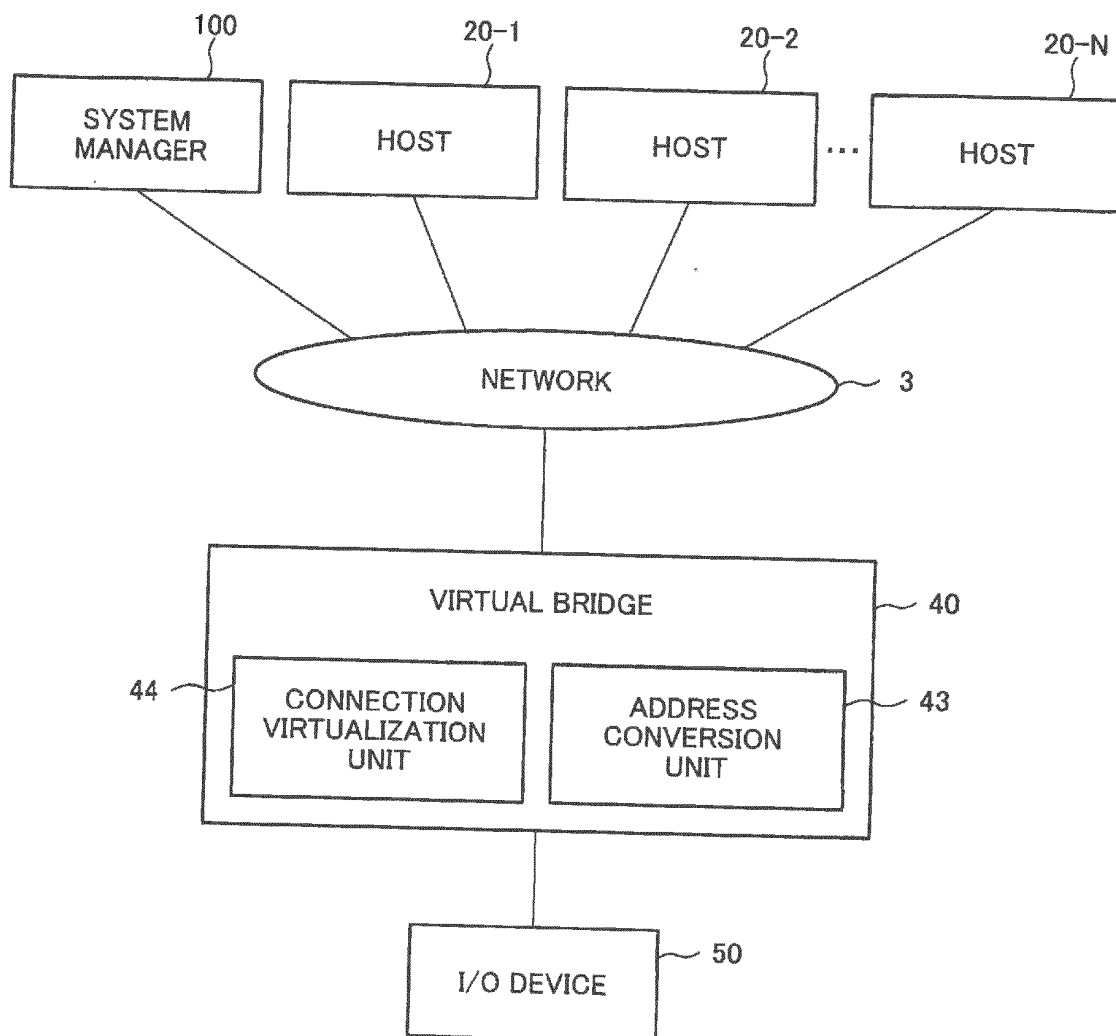


Fig. 6



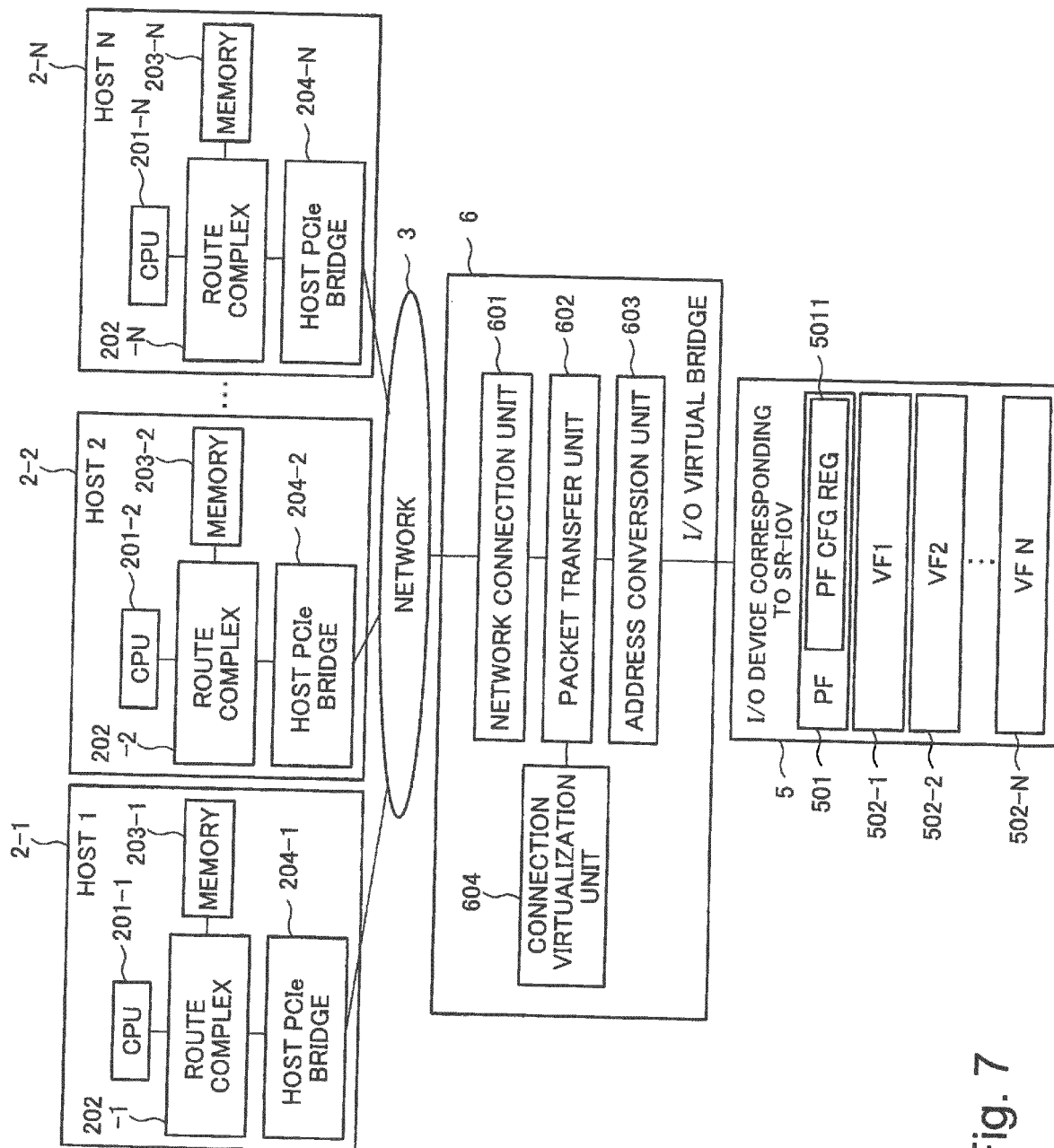


Fig. 7

INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2013/002316

A. CLASSIFICATION OF SUBJECT MATTER

G06F13/10(2006.01)i, G06F3/06(2006.01)i, G06F13/14(2006.01)i, G06F13/36(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F13/10, G06F3/06, G06F13/14, G06F13/36

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Jitsuyo Shinan Koho	1922-1996	Jitsuyo Shinan Toroku Koho	1996-2013
Kokai Jitsuyo Shinan Koho	1971-2013	Toroku Jitsuyo Shinan Koho	1994-2013

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	JP 2010-79816 A (Hitachi, Ltd.), 08 April 2010 (08.04.2010), paragraphs [0019] to [0092]; fig. 1, 2, 4 & US 2010/0082874 A1	1-10
Y	JP 2011-14023 A (NEC Corp.), 20 January 2011 (20.01.2011), paragraphs [0041] to [0078]; fig. 4 (Family: none)	1-10
Y	JP 2012-48546 A (Hitachi, Ltd.), 08 March 2012 (08.03.2012), paragraph [0064] & US 2012/0054393 A1 & EP 2423826 A2	1-10



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"I" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search
19 April, 2013 (19.04.13)Date of mailing of the international search report
14 May, 2013 (14.05.13)Name and mailing address of the ISA/
Japanese Patent Office

Authorized officer

Facsimile No.

Telephone No.

Form PCT/ISA/210 (second sheet) (July 2009)

INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2013/002316

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	JP 2009-301162 A (Hitachi, Ltd.), 24 December 2009 (24.12.2009), paragraphs [0013] to [0028]; fig. 1 & US 2009/0313391 A1	3-5, 8-10

Form PCT/ISA/210 (continuation of second sheet) (July 2009)

REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- WO 2009025381 A [0012]
- JP 2012087888 A [0057]